

# Greedy

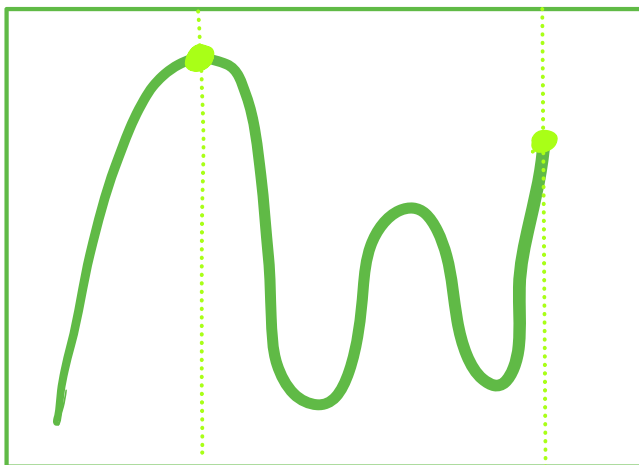
2023.08.24  
동행나 유원빈 2. 그리디 & 구현 시험.

현재 상황에서 **지금 당장 좋은 것**을 고르는 방법.

→ 현재의 해가 맞는지 확인할 수 있어야 함.

## 11501 실버 2 주식 문제풀이 공통

1. 주식 할 게 자기
2. 최대한 만큼 팔기
3. 아무것도 안하기.



주식 그래프

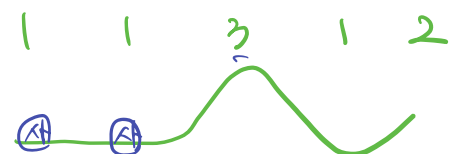
주식 많이 보는 법

= 싸게 사서 비싸게 팔기

→ 최대한 최저가보다 낮은 가격에 사서,  
최대 가격에 팔기.

= 최대한 최저가를 기준으로 생각.

```
for _ in range(int(input())) : # 테스트 케이스 개수
    n = int(input()) # list의 length
    nums = list(map(int, input().split())) # 리스트
    maxNum = nums[-1] # 뒤에서부터 생각하기 중요
    profit = 0
    for i in reversed(range(n)):
        if nums[i] < maxNum :
            profit += maxNum - nums[i]
        elif nums[i] > maxNum:
            maxNum = nums[i]
    print(profit)
```



# 1931 실버① 회의실 문제 개인

· 일찍 끝나는 애들..

## 문제

한 개의 회의실이 있는데 이를 사용하고자 하는 N개의 회의에 대하여 회의실 사용표를 만들려고 한다. 각 회의 i에 대해 시작시간과 끝나는 시간이 주어지고, 각 회의가 겹치지 않게 하면서 회의실을 사용할 수 있는 회의의 최대 개수를 찾아보자. 단, 회의는 한번 시작하면 중간에 중단될 수 없으며 한 회의가 끝나는 것과 동시에 다음 회의가 시작될 수 있다. 회의의 시작시간과 끝나는 시간이 같을 수도 있다. 이 경우에는 시작하자마자 끝나는 것으로 생각하면 된다.

## 입력

첫째 줄에 회의의 수  $N(1 \leq N \leq 100,000)$ 이 주어진다. 둘째 줄부터  $N+1$  줄까지 각 회의의 정보가 주어지는데 이것은 공백을 사이에 두고 회의의 시작 시간과 끝나는 시간이 주어진다. 시작 시간과 끝나는 시간은  $2^{31}-1$ 보다 작거나 같은 자연수 또는 0이다.

```
n = int(input())
rsv = []
cnt = 1
for _ in range(n) : #일찍 끝날 수록 뒤에 회의 채울 수 있음
    s, f = map(int, input().split())
    rsv.append([s, f])
```

```
rsv.sort(key=lambda x:(x[1], x[0]), reverse=False)
```

```
endtime = rsv[0][1]
```

```
for i in range(1, n) :
    if endtime <= rsv[i][0] : # 예약되면?
        endtime = rsv[i][1]
        cnt += 1
```

```
print(cnt)
```

정렬 조건이  
안되면 안됨

일찍 끝나는  
== 일찍 시작했다  
== 뒤에 채워 더 넣을 수 있다.

